

KEMI-TORNION AMMATTIKORKEAKOULU

MQ-valvonnan hallintasovellus

Outokumpu Stainless Oy

Miina Ponkala

Tietojenkäsittelyn koulutusohjelman opinnäytetyö
Web-asiantuntijan suuntautumisvaihtoehto
Tradenomi

TORNIO 2011

TIIVISTELMÄ

Ponkala, Miina. 2011. MQ-valvonnan hallintasovellus. Opinnäytetyö. Kemi-Tornion ammattikorkeakoulu. Kaupan ja kulttuurin toimiala. Tornio. Sivuja 52. Liitteet 1-6.

Tutkimuksen tavoitteena oli toteuttaa sovellus, jolla hallitaan MQ jonomanagereiden valvontajärjestelmää, sekä keskitettyä konfiguraatioiden tallennusta. Tehtävään kuului myös sovelluksen käytön ohjeistus, tarvittaessa koulutus, sekä dokumentaation tekeminen Outokumpu Tornio Worksin käytäntöjen mukaisesti.

Sovelluksen teko eteni vesiputousmallin mukaisesti, vaihe vaiheelta. Sovelluksen toiminnallisuudet toteutettiin PHP-skriptikielellä ja ulkoasu HTML-sivunkuvauskielellä sekä CSS-tyylitiedostoilla.

Tuloksena oli sovellus, joka on kevyt käyttää ja joka palvelee toimeksiantajan tarpeita riittävästi. Ylläpito on helppoa ja jatkokehitys mahdollista ellei jopa suotavaa, sillä kaikkia toiminnallisuuksia en ajan puutteen vuoksi saanut toteutettua.

Mielestäni työ oli kohtuullisen onnistunut pieniä puutteita lukuun ottamatta. Haasteellisinta työssä oli soveltaa tiedoston luku ja – kirjoitusfunktioita työn vaatimalla tavalla.

Asiasanat: Tiedoston hallinta, PHP, CSS, HTML

ABSTRACT

Ponkala, Miina. 2011. Front-end to MQ-monitoring. Bachelor's Thesis. Kemi-Tornio University of Applied Sciences. Business and Culture. Pages 52. Appendices 1-6.

The aim of the thesis was to implement an application that controls the MQ queue managers monitoring, as well as a centralized storage configurations. The assignment also included instructions, training and documentation of the conclusion of Outokumpu Tornio Works' practices.

I used the waterfall method, which means that project proceeded step by step. Application functionality was implemented by the PHP script language and layout by the HTML markup language and cascading style sheet files.

The result was an application that is easy to use and it serves the client's needs adequately. Maintenance is easy and further development possible, even desirable, because I did not have time to do all the required features. I think the work was reasonably successful with the exception of small shortcomings. The most challenging work was found out how to apply the PHP text file functions way the application required.

Keywords: Monitoring, PHP, CSS, HTML

SISÄLTÖ

TIIVISTELMÄ

ABSTRACT

1	JOHDANTO	Virhe. Kirjanmerkkiä ei ole määritetty.
1.1	Opinnäytetyön aihe ja lähtökohta.....	Virhe. Kirjanmerkkiä ei ole määritetty.
1.2	Tutkimuksen tavoite, rajaus ja tutkimusongelmat	6
1.3	Käytettävä tutkimusmetodi	8
1.4	Käytettävät käsitteet ja lyhenteet	9
2	SOVELLUKSEN MÄÄRITTELY	11
2.1	Laitteisto ja ohjelmat.....	11
2.2	Ominaisuudet	11
2.3	Toiminnot.....	11
3	SUUNNITTELU- JA TOTETUSMENETELMÄT	13
3.1	PHP	13
3.2	HTML	14
3.3	CSS.....	14
3.4	MQ API.....	15
3.4	UML.....	16
4	SOVELLUKSEN SUUNNITTELU	18
4.1	Käyttöliittymän suunnittelu.....	18
4.1.1	Asettelu	18
4.1.2	Ulkoasu	19
4.1.3	Navigointi.....	20
4.1.4	Sisältö.....	20
4.2	Moduulit.....	20
4.2.1	Käyttäjän tunnistus.....	20
4.2.2	Navigointimoduuli	21

5	SOVELLUKSEN TOTEUTUS	22
5.1	Käyttöliittymän toteutus.....	22
5.2.	Moduulien toteutus.....	22
6	TESTAUS JA KÄYTTÖÖNOTTO	24
6.1	Testaus.....	24
6.2	Käyttöönotto.....	24
6.2	Ylläpito.....	24
7	JOHTOPÄÄTÖKSET JA POHDINTA.....	26
7.1	Tutkimustulokset.....	26
7.2	Jatkokehitys.....	26
7.3	Pohdinta.....	26
	LÄHTEET.....	28

LIITTEET 1-6

1 JOHDANTO

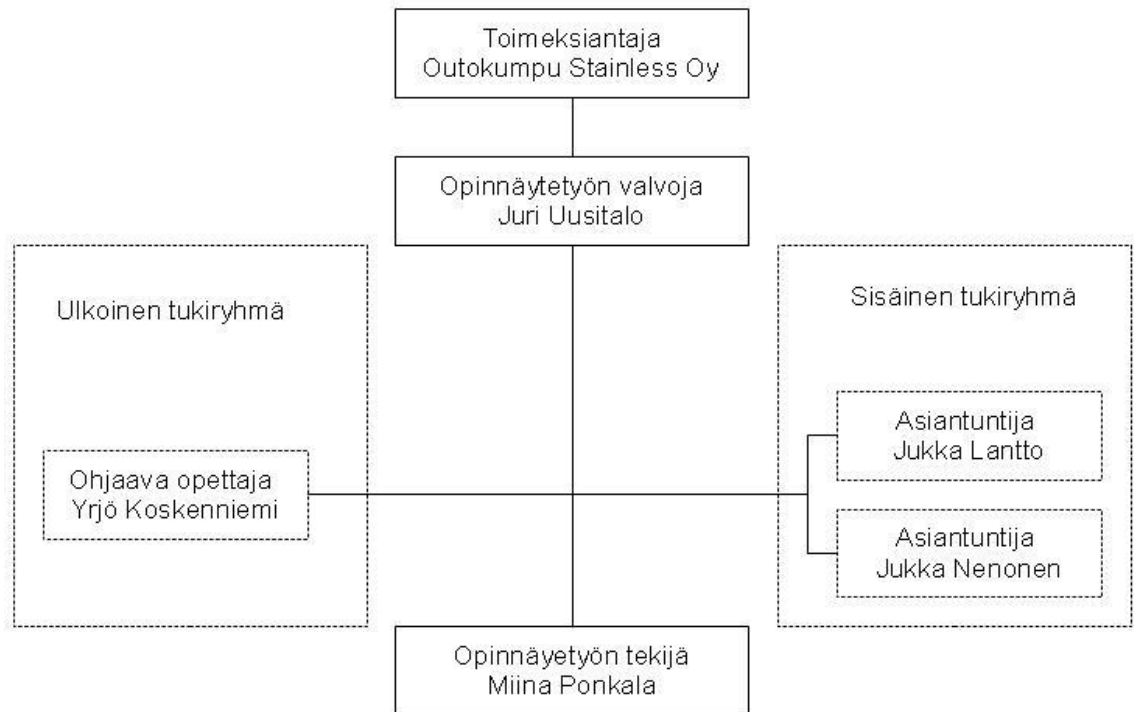
1.1 Opinnäytetyön aihe ja lähtökohta

Tammikuussa 2010 aloitin opintoihini kuuluvan kahdenkymmenen (20) viikon pituisen työharjoittelun Outokummun Tornio Worksissa it-sovellukset-osastolla. Tornio Worksin sovellusosasto käsittää kolme tiimiä, yhteensä n.30 työntekijää.

Jo harjoittelun alkuvaiheessa keskustelin esimieheni kanssa mahdollisuudesta tehdä opinnäytetyö hankkeistettuna Outokummulle. Esimieheni lupasi miettiä sopivia aiheita, mutta muiden kiireiden vuoksi asian käsittely siirtyi. Harjoittelun päätyttyä jatkoin Outokummulla kesäharjoittelijana ja kesän lopulla otin uudestaan puheeksi opinnäytetyön. Esimieheni keskusteli muiden tiiminvetäjien kanssa opinnäytetyöstäni ja he päätyivät Mq-valvonnan hallintasovellukseen, joka on tarpeeksi laaja ja rajattu työ sekä sopii hyvin osaamisalueeseeni. Hallintasovelluksesta tulee selaimella käytettävä sovellus, jonka toiminnallisuudet toteutetaan PHP-skriptikielellä. Ulkoasu toteutetaan HTML-sivunkuvauskielellä ja CSS-tyylitiedostoilla.

Opinnäytetyön lähtökohtana oli asiakkaan selkeä tarve työlle. Outokumpu Stainless Oy:n Tornion tehtailla toteutetaan järjestelmien välinen tiedonsiirto pääosin sanomapohjaisella tekniikalla. IBM:n WebSphere MQ on tuote, jolla tämä tiedonsiirto on toteutettu. Tällä hetkellä tuotantokäytössä on yli 20 jonomanageria eli asennusta ja integrointien toimintavarmuuden takaamiseksi sanomaliikennettä seurataan keskitetysti. Jos sanomaliikenteeseen tulee ongelmia tai häiriöitä, lähetetään asiantuntijoille hälytysviesti. Tämä seuranta on toteutettu Linux-palvelimella ajettavien skriptien avulla. Skriptien hallinta on teknisesti haastavaa, joten selaimella toimivalle käyttöliittymälle, eli web-sovellukselle, on tullut tarvetta. Käyttöliittymällä voidaan helposti lisätä seurattavia jonomanagereita seurantaan. Skripteillä ajetaan myös keskitetysti jonomanagereiden konfiguraatiot talteen säännöllisesti ja myös tämä toiminnallisuus lisätään käyttöliittymään. (Uusitalo 2011.)

Projekti lähti liikkeelle kirjallisuuteen tutustumisella, jota on jatkunut koko tutkimuksen ajan. Aivan alussa nimesimme projektiorganisaation, johon sain Outokummulta ohjaajan ja kaksi asiantuntijaa, jotka määrittelevät työn tavoitteet Outokummun osalta. Seuraava kuva (Kuva 1.) esittää opinnäytetyöprojektin organisaatiokaavion.



Kuva 1. Projektiorganisaatio

Sisäisen tukiryhmän kanssa pidimme aloituspalaverin, jossa suunnittelimme projektin toteutusta ja käytettäviä menetelmiä.

1.2 Tutkimuksen tavoite, rajaus ja tutkimusongelma

Opinnäytetyön tavoitteena on parantaa työn joustavuutta Outokumpu Worksin sovellus- ja infrastruktuuriolosuhteilla. Konkreettisen työn eli sovelluksen perimmäinen tarkoitus on nopeuttaa sanomaliikenteen hallintaa. Tavoitteena on tehdä sovellus, jota kaikki pystyvät käyttämään ja joka pysyy ajan tasalla. Raportoinnin tavoitteena on mahdollistaa tuotteen ylläpito ja mahdollinen jatkokehitys kattavan dokumentaation avulla sekä sovelluksen ulkopuolella pohtia, millaisia vaikutuksia suurille yrityksille on erilaisista sovelluksista ja niiden käytöstä. Sivuan myös hieman it-osaston ja

informaatioteknologian merkitystä suurten yritysten toiminnalle ja tuotannon tekemiselle.

Perinteisesti etenevien tutkimusten ongelmat ovat usein selkeitä ja tarkkaan rajattuja (Hirsjärvi & Remes & Sajavaara 2009, 126).

Tutkimusongelman määrittelyn tueksi selvitin toimeksiantajan mielipiteen sovelluksen ominaisuuksien tärkeydestä. Keskustelun aikana kävi ilmi, että käytettävyydellä on suurin merkitys tuotteen mielekkäässä ja tarkoituksenmukaisessa käytössä.

Tutkimusongelma 1:

Miten saada konfiguraation hallinta pysymään ajan tasalla?

Tutkimusongelma 2:

Miten suunnitella ja toteuttaa selkeä ja helppokäyttöinen sovellus?

Edellä mainitut ongelmat ovat tärkeitä, mille tahansa sovellukselle. Outokummun kokoisessa yrityksessä It-osastolla on suuri merkitys ja on erittäin tärkeää, että käytettävät ohjelmistot, sovellukset ja laitteet ovat toimivia ja ajan tasalla, jotta tuotanto jatkuu häiriöttä. Kehittämäni sovellus ei ole ns. tuotantokriittinen, mutta se on tärkeä apu eri järjestelmien välisen sanomaliikenteen seurantaan, sillä pahimmassa virhetilanteessa voi tuotanto sanomien jumiutuessa ja tukkiessa liikenteen jopa pysähtyä.

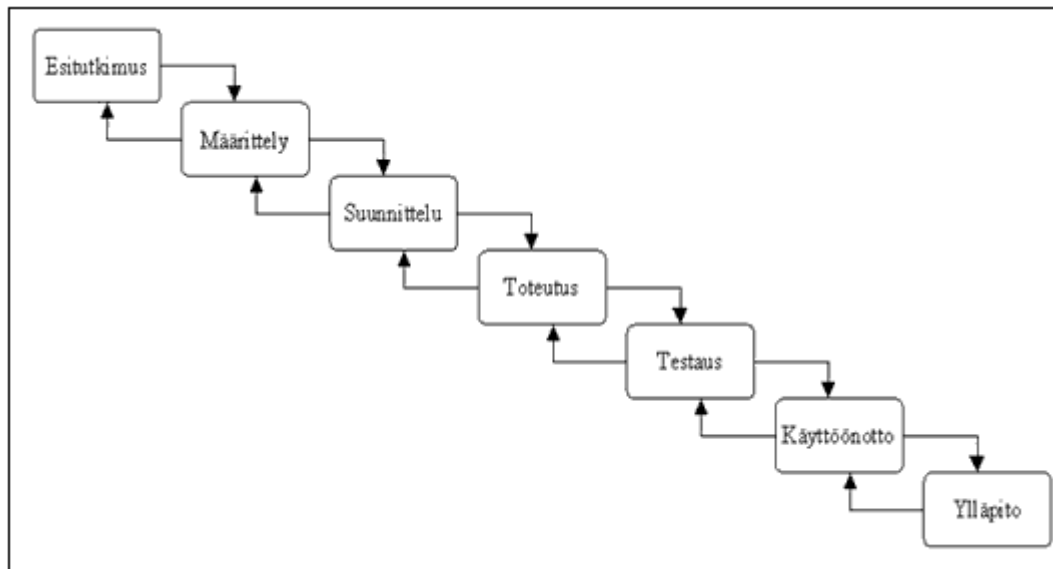
Opinnäytetyön tekemiseen kuuluu esitutkimus, määrittely, suunnittelu ja toteutus sekä testaus ja dokumentointi. Esitutkimus käsitti aloituspalaverin pitämisen toimeksiantajan tukiryhmän kanssa, jossa rajasimme sovelluksen koskemaan konfiguraatietietojen tallentamista sekä hälytystietojen muokkausta. Ajan rajallisuudesta johtuen poistimme sovelluksen toiminnallisuuksista keskitetyn MQ Explorerin jonomanageritiedoston hallinnan ja tallennuksen. Tärkeimmäksi nousi sanomaliikenteen seurantajärjestelmän konfiguraatio-tiedostojen muokkausmahdollisuus, kuten uusien jonojen lisäys ja hallinta. Tutkimuksen lopuksi esittelen työni tulokset ja opastan toimeksiantajaa käytössä.

1.3 Käytettävä tutkimusmetodi

Tutkimusmetodinä käytän konstruktivistista tutkimusta. Se on yleisesti käytetty metodi tietojenkäsittelyn tutkimuksissa, töiden luonteen vuoksi. Suurimmassa osassa tietojenkäsittelyn opinnäytetöissä aiheena on sovellus tai www-sivu.

Konstruktivisessa tutkimuksessa tutkija yrittää luoda jonkin uuden ja paremman ratkaisun entisen tilalle ja siinä korostuu kommunikaation merkitys toimeksiantajan ja tekijän välillä. Tutkimuksessa tarkastelun kohteena voi olla sovelluksen tekemiseen käytetyt tekniikat, työn lopputulos tai tekoprosessi ja siinä käytetyt menetelmät. Konstruktivisessa tutkimuksessa on tärkeää osoittaa, että tutkimus on auttanut luomaan jotain uutta. (Meriläinen, 2010.)

Opinnäytetyön etenemiseen käytän vesiputousmallia, joka on tavallisin vaihejakomalleista. Vaihejaolla tarkoitetaan projektin tai tutkimuksen jakamista vaiheisiin. Vesiputousmalleja on monenlaisia, mutta yleensä ne sisältävät ainakin määrittelyn, suunnittelun ja toteutuksen. Vesiputousmallissa vaiheet etenevät ajanmukaisesti, mutta koska ohjelmistokehityksessä vaatimukset voivat muuttua projektin aikana, sitä ei useinkaan voi täysin tarkkaan seurata, vaan malli toimii ikään kuin suuntaa antavana. (Haikala & Märijärvi 1997, 24-27.) Kuvassa 2 näkyvät vesiputousmallin vaiheet portaikkomaisesti kuvattuna. Nuolet ilmaisevat takaisinpaluun mahdollisuutta.



Kuva 2. Vesiputousmalli

1.4 Käytettävät käsitteet ja lyhenteet

CSS (Cascading Style Sheets) tarkoittaa tyylitiedostoja, joihin määritellään sivuilla esiintyvät tyylit ja HTML-elementtien sijainnit. CSS-tiedostot voidaan linkittää kaikille sovelluksen sisältäville sivuille, joka säästää paljon työtä (CSS Introduction 2011).

HTML on www-sivujen tekoon käytetty kuvauskieli, jolla kuvataan sivujen sisältöä (Santara 1996, 248).

IT on lyhenne, joka tulee englanninkielien sanoista information technology, joka tarkoittaa informaatioteknologiaa, tietotekniikkaa.

Jono on WebSphere MQ -objekti, joka säilyttää sanomia sanomien jonotussovelluksia varten. Jonon hallintaohjelma omistaa jonon ja ylläpitää sitä (IBM® DB2 Universal Database 2011).

Jonomanageri on järjestelmän ohjelma, joka tarjoaa jonotuspalveluja sovelluksissa. MQ API:n avulla ohjelmat voivat lisätä viestejä jonoon tai vastaanottaa niitä. Jono managereita voi olla käynnissä useampi kerrallaan, esimerkiksi tuotannossa ja

testauksessa. Sovelluksessa jokainen jonomanageri tunnistetaan yksilöllisen kahvan (handle) avulla (Uusitalo 2011).

Ohjelmointirajapinta (Application Programming Interface, API) on erityinen menetelmä joka on tietokoneen käyttöjärjestelmän tai sovellusohjelman määrittäjä. Ohjelmoija kirjoittaa määrittäykset sovellukseen, jotta ohjelma voi tehdä pyyntöjä käyttöjärjestelmään tai toiseen sovellukseen. (Li 2000.)

WebSphere MQ on IBM:n lisensoitujen ohjelmien joukko, joka tarjoaa sanomien jonopalveluja (Uusitalo 2011).

2 SOVELLUKSEN MÄÄRITTELY

2.1 Laitteisto ja ohjelmat

Sovelluksen pohjalla on Linuxin tiedostojärjestelmä ja siellä pyörivät palvelut, jotka ottavat yhteyksiä MQ – managereihin MQ – API:n kautta. Http-palvelinohjelmistona on Apache 2.

Kehitysvaiheessa käytän Xampp-ilmaisojelmaa, joka sisältää PHP 5:n ja Apachen. Xampp on avoimen lähdekoodin ohjelma, joka on kehitetty juuri sen takia, että ohjelmoija voi helposti testata tekemisiään paikallisesti omalla koneellaan. Selainpohjaisen sovelluksen tekemisessä siitä on suuri apu.

2.2 Ominaisuudet

Sovelluksen käyttäjinä toimivat Tornio Worksin IT osaston ServiceDesk ja asiantuntijat sekä sähkökunnossapidon vuorohuoltoteknikot. Käyttöliittymän tulee toimia normaalisti sekä Internet Explorer- että Mozilla Firefox -selaimilla. Käyttöliittymän käyttö ei vaadi mitään muuta kuin Internet- tai Intranet-yhteyden, kirjautumisen ja toimivan selaimen.

2.3 Toiminnot

Sovelluksen toiminnallisuudet voidaan jakaa kahteen ryhmään. Ensimmäisessä on ini-tiedostojen muokkaus, johon käyttäjä voi syöttää haluamansa tiedon, poistaa tai muokata sitä. Tätä toimintoa toteutetaan lomakkeella, johon on haettu oletusarvot suoraan valitusta tiedostosta. Tähän muokkaus – näkymään on sisällytetty myös mahdollisuus luoda kokonaan uusi ini-tiedosto valmiin mallin mukaisesti.

Toinen toiminnallisuus koskee jonomanagereiden seuranta ja niiden valvomista. MQ API-laajennuksen avulla palvelimelta haetaan kaikki käytössä olevat jonomanagerit ja

niiden statukset. Tämä toteutetaan MQ API:n avulla. MQ API:lla voidaan lisätä tai hakea viestejä jonojen jakelulistoille. (PHP Manual 2011).

3 SUUNNITTELU- JA TOTEUTUSMENETELMÄT

3.1 PHP

PHP eli Hypertext Preprocessor on ohjelmoimiseen käytettävä palvelinpuolen skriptikieli. Sitä käytetään usein HTML-kuvauskielen kanssa ja se antaa mahdollisuuden melkein mihin vain. PHP oli alun perin suppea kieli, jonka avulla pystyi ylläpitämään omia kotisivujaan. Ominaisuuksien lisääntyessä PHP:n suosio kasvoi ja siitä tuli yksi suosituimmista ohjelmointikielistä.

PHP-koodia tehtäessä, tulkille täytyy kertoa, että on käännettävä PHP-koodia. Koodi kirjoitetaan tavallisella tekstieditorilla. Koodi aloitetaan aloitustagilla `<?php` ja lopetetaan lopetustagilla `?>`. Näiden merkkien ulkopuolella oleva teksti tulostuu selaimella HTML-koodina. (Zandstra 2000, 20, 36–37.)

Palvelin käsittelee ja tulkkaa PHP-koodin ennen lähettämistä selaimelle, minkä vuoksi se ei vaadi selaimelta tukea. PHP:llä voidaan käsitellä myös tietokantoja ja palvelimen tiedostoja. (Laaksonen, 2011).

Kuvassa 3 PHP-koodi on upotettu html:n sekaan ja selain näyttää tekstin ”Hello World” lihavoituna.



```
1 <html>
2 <head>
3   <title>PHP-esimerkki</title>
4 </head>
5 <body>
6   <b>
7     <?php
8       print "Hello World";
9     ?>
10  </b>
11 </body>
12 </html>
```

Kuva 3. Esimerkki PHP-koodista

Opinnäytetyössä toteutan PHP:llä kaikki sovelluksen toiminnallisuudet. Apuna käytän MQ Apia, joka on rajapinta PHP:n ja MQ-tiedostojen välillä. Tässä sovelluksessa ei pohjana ole tietokantaa, vaan muutettavat tiedot lisätään suoraan tiedostoihin.

3.2 HTML

HTML (Hypertext Markup Language) on www-sivujen kuvauskieli. Se sitoo sisällön, grafiikan ja muun informaation yhteen. HTML-koodi on vain tekstiasiakirja, jonka päätteeksi talletetaan muotoon .htm tai .html. Sen voi aukaista suoraan selaimella. HTML-kieli on yleismaailmallista ja toimii samalla tavoin sekä Windows- että Mac-koneilla. Mitä tahansa ohjelmointikieltä sivujen tekoon on käytetty, sen pohjalla on aina HTML:ää. (Noble & Tittel 2008, 10).

HTML-koodi koostuu tunnisteista ja elementeistä ja tunnisteille annetuista attribuuteista. Tunnisteet merkitään aina pareittain ja tunnisteparin väliin laitetaan elementti. Tunniste avataan pienempi kuin -merkillä ja suljetaan suurempi kuin -merkillä. Lauseen lopettavan parin eteen laitetaan vielä /-merkki. Ennen tunnisteiden sulkemista lisätään attribuutit. Esimerkkikoodissa, joka näkyy kuvassa 4, määritellään otsikko ja kappale. Kappaleen fontin väriksi määritellään punainen. ¨- merkki kappaleen elementissä eli sanassa leipäteksti merkitsee ä-kirjainta. Käyttämällä universaalia merkistöä varmistetaan ö- ja ä-kirjainten oikein näkyminen, missä tahansa sivuja luetaan.

```

1  <html>
2    <head><title>Esimerkki</title></head>
3  <body>
4    <h1>Otsikko</h1>
5    <p style='color:red;'>Leip&uml;teksti</p>
6  </body>
7  </html>

```

Kuva 4. Esimerkki HTML-koodista

3.3 CSS

CSS:ää (Cascading Style Sheets) käytetään tyylimuotoiluun. Se on universaali, kuten HTML-kielikin. CSS:n avulla kerrotaan web-sivuille, miltä niiden tulisi näyttää. On suositeltavaa, että kaikki HTML-sivuilla käytettävät tyylit kirjoitetaan CSS-tiedostoon, joka linkitetään HTML-sivulle. (Noble & Tittel 2008, 10.)

CSS voi näyttäytyä erilaisina eri selaimilla, joten tekstitiedoston on oltava validia, jotta näkyvyys olisi samanlainen selaimesta riippumatta. Kuvassa 5 määritellään tyylitiedostossa sivun rungon taustakuva sekä minimi- ja maksimileveys. Otsikko 3:een määritetään fontti, fontin koko, väri sekä pakotetaan kaikki kirjaimet kapitaaleiksi. Top- ja left-attribuuteilla määritetään otsikon sijainti.

```

1  body
2  {
3
4      background-image: url(images/body-background.jpg) ;
5      background-position: top left;
6      background-repeat: no-repeat;
7      min-width: 500px;
8      max-width: 1244px;
9  }
10 h3
11 {
12     font: 16px arial, helvetica, sans-serif;
13     font-weight: bold;
14     color: #000066;
15     text-transform: uppercase;
16     top: 18.5%;
17     left: 49%;
18 }
```

Kuva 5. Esimerkki CSS-tyylitiedostosta

3.4 MQ API

MQ API on ohjelmointirajapinta, joka sisältää hyödyllisiä funktioita ohjelmoinnin tueksi. PHP:lle tarkoitettu laajennus on nimeltään Mqseries ja sen avulla voi lisätä viestejä jonoon, hakea viestejä jonosta sekä paljon muuta. (Tjon-A-Hen 2011). Kuvassa 6 on esimerkki **mqseries_inq()** – funktiosta, joka palauttaa joukon kokonaislukuja sekä sarjan merkkijonoja jotka sisältävät objektin ominaisuuksia.


```

1  <?php
2      $int_attr = array();
3      $char_attr = "";
4
5      mqseries_inq($conn, $obj, 1, array(MQSERIES_MQCA_Q_MGR_NAME), 0,
6          $int_attr, 48, $char_attr, $comp_code, $reason);
7
8      if ($comp_code != MQSERIES_MQCC_OK) {
9          printf("INQ CompCode:%d Reason:%d Text:%s<br>\n",
10              $comp_code, $reason, mqseries_strerror($reason));
11      } else {
12          echo "INQ QManager name result ".$char_attr."<br>\n";
13      }
14  ?>

```

Kuva 6. Mqseries-funktio

Mqseries – laajennuksen tavoitteena on tarjota käyttöliittymä, joka kommunikoi sovelluksen kautta käytössä olevien jonomanagereiden kanssa. Käyttöliittymä jäljittelee mahdollisimman tarkasti C-kielelle tehtyä laajennusta, jota IBM Websphere MQ käyttää. Käytössä ovat samat nimeämiskäytännöt ja mahdollisuudet. Laajennuksen ymmärtämiseksi tarvittaisiin jonkinlaista tietämystä C-API:sta, jota minulle ei ole. Tämä tekee Mqseriesin käytöstä haastavan. (PHP manual 2011.)

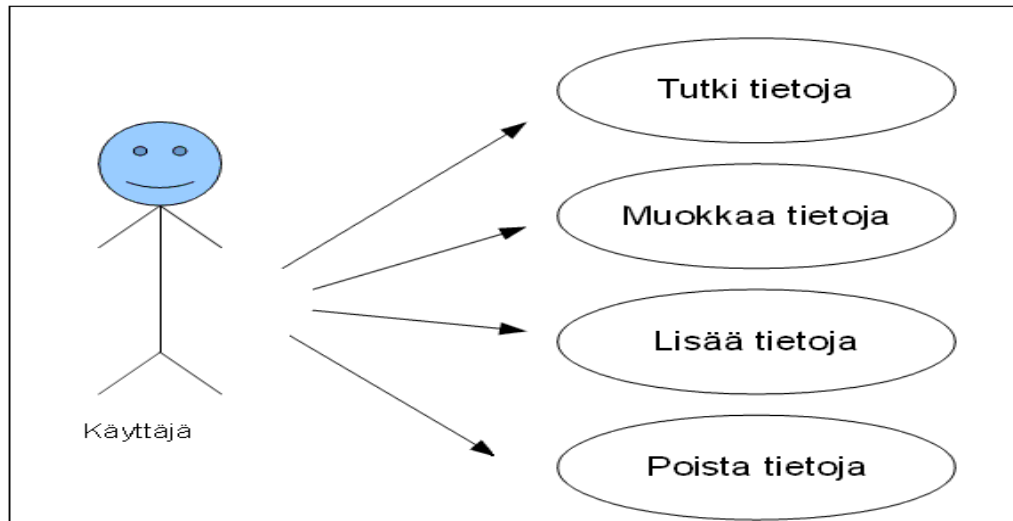
3.5 UML

UML eli Unified Modeling Language, on mallinnuskieli, jolla havainnollistetaan sivuston tai yksittäisen moduulin tai järjestelmän toimintaa. UML toimii kommunikaatiovälineenä koneen ja ihmisen välillä.

UML:n avulla voidaan ratkaista ongelmatilanteita tai laatia dokumentteja. Suosituimpia UML kaavioita ovat ER-kaaviot, tietovuokaaviot, lohkokaaviot ja tilakaaviot. Käyttötapauskaavio on yksi mallinnuskaavio, joka yksinkertaisimmallaan kuvaa käyttäjän toimintaa sovelluksella. Kehitysvaiheessa käyttötapauskaaviolla on helppo jäsenellä mitä toiminnallisuuksia sovelluksen käyttö vaatii.

(Koskimies & Koskinen & Maunumaa & Peltonen & Selonen & Siikarla & Systä 2011).

Kuvassa 7 esitellään yksinkertainen käyttötapauskaavio sovelluksen toiminnoista, joita käyttäjä voi tehdä.



Kuva 7. Käyttötapauskaavio

4 SOVELLUKSEN SUUNNITTELU

4.1 Käyttöliittymän suunnittelu

Käyttöliittymän suunnitteluvaiheessa tulee ottaa huomioon asiakkaan vaatimukset käyttöliittymän selkeydestä ja helppokäyttöisyydestä. Peruseriaate sovelluksessa on, että muokkaus onnistuu syöttämällä tiedot lomakkeelle, joka tallentaa annetut arvot suoraan tiedostoihin. Lomakkeen tulisi sisältää myös mahdollisuus hakea jo olemassa olevia tietoja muokkausta varten.

Ulkoasulle ja sivujen asettelulle sain toimeksiantajalta vapaat kädet. Nettisivuja ja käyttöliittymiä on maailmassa kuitenkin tehty ja suunniteltu jo vuosia, jonka vuoksi on tiettyjä käsityksiä siitä, mikä toimii ja mikä ei. (Korpela & Linjama 2003, 360.)

4.1.1 Asettelu

Sivun asettelun lähtökohdaksi otin selkeyden ja oman kokemukseni miellyttävästä käyttöliittymästä. Sivun asettelu koostuu kolmesta elementistä, joita ovat ylätunniste, alatunniste ja itse sisältö. Kuva 5 havainnollistaa sivun rakenteen.



Kuva 7. Sivun asettelu

4.1.2 Ulkoasu

Korpelan & Linjaman (2003, 362) mukaan näytettävyyden ei ole se käyttäjän huomion kiinnittävä tekijä, vaan tietosisältö ja käytettävyyden. Olenkin pyrkinyt tekemään sivuston ulkoasun mahdollisimman hillityksi ja yksinkertaiseksi.

Käytettävyyden perustuu usein visuaaliseen ulkomuotoon ja siihen, mitä silmä on tottunut näkemään. Helppokäyttöisyyttä mitataan esimerkiksi sillä, miten nopeasti uusi käyttäjä oppii käyttämään sovellusta ja käyttö muodostuu rutiiniksi. Tärkeää käytettävyyden kannalta on etusivu, joka antaa mielikuvaa koko sovelluksesta. Etusivun on oltava selkeä ja navigoinnin on oltava selkeästi esillä. (Korpela & Linjama 2003, 360.)

Väreinä käytän Outokummun intranetistä tuttuja värejä ja ylätunnisteessa on Outokummun logo, joka on myös kaikilla Outokummun intranet-sivuilla. Sisältöä ei sovelluksessa ole kovin paljon, joten se sijaitsee näytön keskellä, jottei sivun oikealle puolelle jäisi liikaa tyhjää tilaa. Visuaalisen ilmeen parantamiseksi olen logon lisäksi käyttänyt Monitoring -sivulla kuvia. Käytetyt kuvat ovat ennestään tuttuja ja turvallisia, kuten muokkaa-linkissä kynä, poista-linkissä roskakori. Hiiren ollessa kuvan päällä, näkyy kuvan alt-tekstinä linkin tarkoitus.

4.1.3 Navigointi

Navigointi sivulla on yksikertainen siitäkin syystä, ettei varsinaisia linkkejä ole kuin kaksi. Olen asetellut ne ylätunnistimeen, jolloin ne näkyvät kaikilla sivuilla.

Muokkausvalikkoon siirrytään tiedoston nimen vieressä olevista kuvakkeista. Ne ovat kaikkien tiedostonimien vieressä siistissä rivissä, joka on miellyttävän näköinen ja helppo käyttää. Jokaisella lomakkeen tai muokkausvalikon sisältävällä sivulla on Tallenna-painikkeen vieressä Peruuta-painike, jonka avulla pääsee helposti takaisin edelliselle sivulle

4.1.4 Sisältö

Sisältö on sovelluksessa dynaaminen, mikä tarkoittaa, että sisällön koko riippuu jonomanagereiden määrästä. Jos jonoja on paljon, kasvaa sisältöelementti pituussuunnassa. Sisältö koostuu taulukosta, johon listautuu kaikki tarvittavat tiedostot ja muokkauslomakkeesta. Muokkauslomakkeelle päästään klikkaamalla tiedoston nimen vieressä olevaa kynä-ikonia.

4.2 Moduulit

4.2.1 Käyttäjän tunnistus

Turvallisuussyistä sovellusta käyttävät vain ennalta määrätyt käyttäjät. PHP:n ja tietokantojen avulla voi monella tapaa toteuttaa salasana-kyselyjä, mutta koska tällä sovelluksella ei ole tietokantaa pohjana, salasanasuojaus toteutetaan palvelimen avulla. Itse en tehnyt sovellukseen käyttäjän tunnistus – moduulia, vaan se hoidetaan Service Deskin kautta Outokummun standardien mukaisesti.

4.2.2 Navigointimoduuli

Navigointi on sijoitettu omaan paikkaansa ylätunnisteessa. Ylätunnisteen idea on, että sinne kirjoitetaan kaikki HTML:n metatiedot ja HTML-dokumentin aloitustagit sekä linkki CSS-tiedostoon. PHP:n avulla ylätunnisteen voi helposti liittää kaikille käytettäville sivuille, joten siinä olevia tietoja ei tarvitse erikseen lisätä jokaiselle sivulle. Tämän vuoksi sijoitin yläpalkkiin myös navigaation, jolloin se seuraa mukana sivulta sivulle, eikä sen sijainti muutu.

5 SOVELLUKSEN TOTEUTUS

5.1 Käyttöliittymän toteutus

Jo työn alkuvaiheessa päätin, että käyttöliittymän toteutus tulee olemaan mahdollisimman yksinkertainen. Yksinkertainenkin voi miellyttää silmää ja sen hyvä puoli on, että osien latautumiseen ei mene turhaan aikaa. Työharjoittelussani tein Tornion tutkimuskeskukselle intranet-sivuston. Sain intranetin ulkoasusta hyvää palautetta, joten otin siitä ideoita opinnäytteeksi tehtävään sovellukseen. Fontit, värit ja navigaatio noudattavat samaa linjaa intranetin kanssa.

Käyttöliittymän toteutukseen käytin HTML-sivunkuvauskieltä. Käyttöliittymän sisältö koostuu lähinnä taulukoista ja lomakkeista. Initietojen muokkauslomakkeessa käytin HTML fieldset-tagia, joka sitoo lomakkeen kentät loogiseksi ryhmäksi piirtämällä laatikon niiden ympärille. Legend-tagilla voidaan antaa nimi lomakekentät sisältävälle laatikolle.(w3schools, 2011.) Fieldsetin käyttö antaa mielestäni siistin vaikutelman ja on tuttu myös käyttöliittymien, kuten Windowsin toiminnoista.

Kaikki käyttöliittymän tyylit on määritelty yhteen tyylitiedostoon, joka on liitetty navigointimoduuliin. CSS-tiedosto sisältää taulukoiden, fieldsetin, otsikkojen ja navigaatiopalkin tyylimäärittelyt.

5.2 Moduulien toteutus

Outokummun käytäntöjen mukaisesti tietyt sisällöt ja näkymät on rajattu virheiden ja väärinkäytön välttämiseksi ryhmille.

Navigointimoduuliin otin mallia tekemästäni intranetsovelluksesta. Päätin käyttää siinä hyväksi havaitsemaani CSS-muotoiltua alasvetovalikkoa. Tässä sovelluksessa valikko on hyvin minimaalinen, koska se sisältää vain kaksi otsikkotason linkkiä, eikä yhtään alatasoa. Muutoksia ajatellen päädyin kuitenkin tähän, koska siihen on helppo lisätä uusia tasoja, jos tarvetta ilmaantuu.

Valikon lisäksi navigointimoduulissa on Outokummun logo, jota käytetään kaikissa Outokummun selainpohjaisissa sovelluksilla sekä internet- ja intranetsivuilla. Logon alapuolelle näkyy PHP:llä toteutettu päivämäärän tulostus. Kuvassa 8 näkyy navigointimoduuli kokonaisuudessaan.



Kuva 8. Navigointimoduuli

6.1 Testaus

Ohjelmistojen testauksessa pyritään löytämään virheet ja niiden etsintään voidaan tehdä tarkoituksen mukaisia testitapauksia, jolloin säästetään aikaa verrattuna satunnaiseen kokeiluun ja sen kautta virheiden löytämiseen. (Haikala & Märijärvi 1997, 230.)

Sovelluksen luonteen vuoksi päätin pohjata testauksen käytettävyydestestaukseen (usability testing), jossa kehittäjä tekee testitapauksia, joita testaaja pyrkii noudattamaan vaihe vaiheelta. (Haikala & Märijärvi 1997, 236.)

Määrittelyvaiheessa testasin itse sovellusta ja lopullinen testaus suoritettiin Tornio Worksin testiympäristössä Sovin testauspäivän ohjaajani kanssa ja tein etukäteen muutaman testitapauksen, jotka suunnittelin sovelluksen toiminnallisuuksien perusteella.

Testausta oli suorittamassa itseni lisäksi ohjaajani Outokummulla. Testauksen perusteella kävi ilmi, että testatut toiminnot ja moduulit toimivat tarkoituksenmukaisesti.

Testitapaukset tuloksineen näkyvät liitetiedostoissa.

6.2 Käyttöönotto

Sovellusten ja muutosten käyttöönotolle on Tornio Worksilla omat standardinsa. Käyttöönotoista ilmoitetaan tietyssä aikataulussa ja tietyille ihmisille. Aikatauluista johtuen, en ollut paikan päälle käyttöönoton aikana.

6.3 Ylläpito

Ylläpitoa varten tein tarvittavat dokumentit ja ohjeet, sekä kommentoin koodini riittävästi. Niiden perusteella jokainen sisäisestä asiantuntijaryhmästäni voi ylläpitää ja päivittää sovellusta tarpeiden mukaan.

7.1 Tutkimustulokset

Tutkimustuloksena syntyi käyttöliittymä, jolla pystyy muokkaamaan asiakkaan haluamia tietoja selaimella. Konfiguraatiomuokkauksessa tiedosto tulostuu sellaisenaan tekstikenttään ja sitä voi vapaasti muokata. Initiedostoja muokataan lomakkeella, johon tulostuu oletuksena olemassa olevat tiedot. Kenttien syöttöön on annettu ohjeet ja virheellisistä syötöistä tulee ilmoitus, eikä lomaketta voi tallentaa.

Sovellus on erittäin helppokäyttöinen ja sen ylläpitoon ei vaadita paljoa. Sovellukseen on sisällytetty vain välttämätön tieto ja lähdekoodit on kommentoitu ja nimetty kuvaavasti, joka helpottaa ylläpitoa.

Navigointi on yksinkertainen ja kaikilta sivuilta pääsee takaisin alkuun klikkaamalla navigointipalkissa olevaa Outokummun logoa. Navigointipalkki on täysin CSS-tyylipohjilla muotoiltu drop-down valikko. Tällä hetkellä navigointiin kuuluu vain kaksi pääotsikkoa, mutta jos sovellusta jatkokehitetään ja sivuja tulee lisää, voidaan navigointipalkkiin helposti lisätä alaotsikoita ja niille alaotsikoita.

7.2 Jatkokehitys

Valmistuneen sovelluksen kehitystä on helppo jatkaa. Jonojen hallintaan käytettävää MQ API:a en ehtinyt sisällyttää sovellukseen, joten se lienee pääasia jatkokehityksessä. Tällä hetkellä sovelluksessa voidaan valita mitä jonoja lisätään tiedostoihin, mutta MQ API:n avulla jonoja voidaan hallinnoida valitsemalla suoraan tuotannossa olevista jonoista halutut. Jonoille on tarkoitus tallentaa henkilökohtaiset hälytystiedot, joka mahdollistuu myös MQ API:n kautta.

7.3 Pohdinta

Tutkimuksen alkuperäistä aikataulua ei pystytty täysin toteuttamaan erinäisten teknisten vikojen vuoksi, mutta vaiheet ovat kuitenkin pysyneet samana. Projektin toteuttamisvaihe oli vaativin ja vei eniten aikaa. Jälkikäteen ajateltuna aikaa

opinnäytetyön tekemiseen oli varattu liian vähän. Ajan puutteeseen vaikutti aiheen myöhäinen valinta ja olettamukseni opinnäytetyön aikataulusta. Kesätoissa ollessani ajattelin aikaa olevan 4 kuukautta, syyskuusta joulukuuhun. Henkilökohtaistenkin syiden vuoksi aikaa opinnäytetyön tekemiseen oli lopulta vain kaksi kuukautta.

LÄHTEET

Painetut

- Haikala, Ilkka & Märijärvi, Jukka 1997. Ohjelmistotuotanto. 3. Painos. Gummerus Kirjapaino Oy, Jyväskylä.
- Hirsjärvi, Sirkka & Remes, Pirkko & Sajavaara, Paula 2009. Tutki ja kirjoita. 15. uudistettu painos. Kariston Kirjapaino Oy, Hämeenlinna.
- Korpela, Jukka K. & Linjama, Tero 2003. Web-suunnittelu. Docendo Finland Oy, Jyväskylä.
- Noble, Jeff & Tittel, Ed 2008. HTML, XHTML and CSS For Dummies. 6. painos. Wiley Publishing Inc, Hoboken, NJ, USA.
- Santara, Jouni 1996. WWW-kehittäjän opas. Gummerus Kirjapaino Oy, Jyväskylä.
- Zandstra, Matt 2002. PHP Trainer Kit. 2. painos. Edita Prima Oy, Helsinki.

Painamattomat

CSS Introduction. Luettu 5.12.2011. http://www.w3schools.com/css/css_intro.asp

IBM® DB2 Universal Database Sanasto. Luettu 12.11.2011.

[<http://public.dhe.ibm.com/ps/products/db2/info/vr82/pdf/fi_FI/db2t0y81.pdf>](http://public.dhe.ibm.com/ps/products/db2/info/vr82/pdf/fi_FI/db2t0y81.pdf)

Koskimies, Kai & Koskinen, Johannes & Maunumaa, Mika & Peltonen, Jari & Selonen, Petri & Siikarla, Mika & Systä, Tarja. UML työvälineenä ja tutkimuskohteena. Luettu 28.11.2011

[<http://www.cs.tut.fi/~ohar/kirjallisuutta/UML%20tyovalineena%20ja%20tutkimuskohdeena.pdf>](http://www.cs.tut.fi/~ohar/kirjallisuutta/UML%20tyovalineena%20ja%20tutkimuskohdeena.pdf).

Laaksonen, Antti. 2011. PHP-ohjelmointi: Osa 1 – Johdanto. Luettu 27.10.2011.

[<http://www.ohjelmointiputka.net/oppaat/opas.php?tunnus=php_01>](http://www.ohjelmointiputka.net/oppaat/opas.php?tunnus=php_01).

Li, Thomas 2000. SearchExchange. Luettu 24.11.2011.

[<http://searchexchange.techtarget.com/definition/application-program-interface>](http://searchexchange.techtarget.com/definition/application-program-interface).

Meriläinen, Juha 2010. Konstruktiivinen tutkimus. Luentomoniste. Kemi-Tornion Ammattikorkeakoulu. Kaupan ja kulttuurin toimiala.

PHP manual. Mq series. Luettu 10.11.2011.

[<www.php.net/manual/en/intro.mqseries.php>](http://www.php.net/manual/en/intro.mqseries.php).

Uusitalo, Juri 2011. Projektisuunnitelma. 30.8.2011.

Tjon-a-hen, Philippe

w3schools. 2011. HTML <fieldset> Tag. Luettu 17.11.2011.

[<http://www.w3schools.com/tags/tag_fieldset.asp>](http://www.w3schools.com/tags/tag_fieldset.asp).

W3schools. 2011. CSS Introduction. Luettu 20.11.2011.

[<http://www.w3schools.com/css/css_intro.asp>](http://www.w3schools.com/css/css_intro.asp).

LIITTEET

VAATIMUSMÄÄRITTELY

1. Opinnäytetyön tavoite

Tavoitteena on toteuttaa sovellus, jolla hallitaan MQ jonomanagereiden valvontajärjestelmää, sekä keskitettyä konfiguraatioiden tallennusta. Sovelluksen käyttö tulee ohjeistaa, tarvittaessa kouluttaa, sekä dokumentoida järjestelmä tilaajan käytäntöjen mukaisesti.

2. Tutkimuskysymykset

Tilaaajalle tärkeintä on saada toimiva jonomanagereiden hallinta ja keskitetty, ajan tasalla oleva, konfiguraation hallinta. Sovelluksen käytettävyyteen ja selkeyteen tulee kiinnittää erityistä huomiota.

3. Työ- ja tutkimusmenetelmät

a. Esitutkimus

- i. Asiakkaalla oli selkeä tarve työlle

b. Määrittely

- i. käydään läpi valvontaskriptien MQ-managerikonfiguraatiot ja suunnitellaan niiden pohjalta ajonäytöt
- ii. tutustutaan skriptien ajamiseen ja suunnitellaan niiden ajonäytöt
- iii. keskitetty MQ-konfiguraatioiden tallennus
- iv. keskitetty MQ Explorerin jonomanageritiedoston hallinta ja tallennus (optio)

c. Suunnittelu

- i. Käytetään ketteriä menetelmiä

d. Toteutus

- i. Käytetään ketteriä menetelmiä

e. Testaus

- i. Suunnitellaan testitapaukset toiminnallisuuksien perusteella
- ii. Testataan Tornio Worksin testiympäristöissä

f. Käyttöönotot

- i. Suunnitellaan ja aikataulutetaan käyttöönotto Tornio Worksin standardin mukaisesti.
 - g. Ylläpito
 - i. Ylläpitoa varten dokumentoidaan sovelluksen toiminnallisuudet
 - ii. Sovelluksen koodi on kommentoitava riittävästi.
- 4. Johdanto
 - a. Tuote ja ympäristö
 - i. IBM WebSphere MQ, RedHat Linux 5, Apache 2, PHP 5
 - ii. Ajettavat skriptit on koodattu C++:lla
 - b. Määritelmät, termit ja lyhenteet
 - i. MQ, manageri, jono
- 5. Yleiskuvaus
 - a. Toiminta
 - i. Aukaistaan työn tekemisen yhteydessä, ks. 3b
 - b. Käyttäjät
 - i. Tornio Worksin IT-osaston ServiceDesk ja asiantuntijat, sähkökunnossapidon vuorohuoltoteknikot.
- 6. Tiedot ja tietokanta
 - a. Tietosisältö
 - i. Katso 3b
 - b. Kapasiteettivaatimukset
 - i. Levytilaa konfiguraatioiden tallennukselle muutama mega.
 - c. Tiedostot ja asetustiedostot
 - i. Tiedostot sijaitsevat serverin web-palvelun hakemistoissa.
 - ii. Sovelluksen lähdekoodit tallennetaan versionhallintaan.
- 7. Ulkoiset liittymät
 - a. Laitteistoliittymät
 - i. Ei ole.
 - b. Ohjelmistoliittymät
 - i. Sovelluksella ajetaan Linuxin shell-skriptejä.
 - ii. MQ API (PHP-moduuli)
 - c. Tietoliikenneliittymät
 - i. MQ API:lla
- 8. Muut ominaisuudet
 - a. Saavutettavuus ("käytettävyys"), toipuminen, turvallisuus, suojaukset
 - i. Sovellus tehdään websivuksi.
 - b. Siirrettävyys ja yhteensopivuus
 - i. Valvonta- ja skriptienluontisovellus on alustariippumaton

ii.

iii. Ajastettujen ajojen luonti on alustariippuvainen, tässä tapauksessa sovellus tulee pyörimään Linuxin päällä

c. Operointi

i. Sovellus on websovellus, jota operoivat IT-asiantuntijat.

Mq-valvonnan hallintasovellus, test-case-01:

Purpose:	Avataan konfiguraatietiedot ja muokataan niitä.
Prereq:	Käyttäjä on saapunut etusivulle.
Test Data:	Confsaver kansioista conf.txt tiedosto
Steps:	<ol style="list-style-type: none"> 1. Käyttäjä valitsee navigoinnista Confsaver. 2. Näkee listassa muokattavan tekstitiedoston. 3. Valitsee muokkaa-linkin. 4. Tulee muokkaustilaan. 5. Muokkaa tiedostoa ja tallentaa sen. 6. Palaa takaisin aloitussivulle.
Expected results	Käyttäjä näkee listattuina tiedostot joita on mahdollista muokata ja pystyy muokkaamaan niitä. Muutetut tiedot tallentuvat palvelimelle.
Test results (date/result/tester)	<p>8.12.2011</p> <p>Conf.txt tiedostoa pystyi muokkaamaan normaalisti ja tiedot tallentuivat palvelimelle.</p> <p>Miina Ponkala</p>

Mq-valvonnan hallintasovellus, test-case-02:

Purpose: Luodaan uusi tiedosto monitoring-näkymään.

Prereq: Käyttäjä on saapunut etusivulle.

Test Data: inc kansiota funktiot.php tiedosto

- Steps:**
1. Käyttäjä valitsee navigoinnista Monitoring.
 2. Näkee listan alapuolella Luo uusi tiedosto-linkin.
 3. Valitsee linkin.
 4. Tulee uuden tiedoston luontitilaan.
 5. Lisää tarvittavat tiedot ja painaa tallenna.
 6. Palaa takaisin aloitussivulle.

Expected results Käyttäjä pystyy tallentamaan uuden initiedoston valituilla tiedoilla. Uusi tiedosto tallentuu palvelimelle käyttäjän valitsemaan kansioon.

**Test results
(date/result/tester)** 8.12.2011

Uuden tiedoston luonti ei onnistunut halutulla tavalla.

Koodiin tehtyjen muutoksien jälkeen tallennus onnistui.

Miina Ponkala

Mq-valvonnan hallintasovellus, test-case-03:

Purpose: Tallennetaan lomake oikein täytettyinä.

Prereq: Käyttäjä on saapunut etusivulle.

Test Data: Inifiles

- Steps:**
1. Käyttäjä valitsee navigoinnista Monitoring.
 2. Valitsee muokkaa-linkin.
 3. Tulee muokkaustilaan.
 4. Jättää vuorotellen kenttiä tyhjäksi.
 5. Yrittää tallentaa lomakkeen.
 6. Kirjoittaa tekstiä numeerista tietoa vaativiin kenttiin. (Alarm limit ja Timestamp)
 7. Yrittää tallentaa lomakkeen.

Expected results Jos kenttiä on jätetty tyhjäksi tai täytetty väärin, tulee virheilmoitus eikä pystytä tallentamaan lomaketta. Lomake tallentuu vain oikein täytettynä.

Test results (date/result/tester) 8.12.2011
Mitään kenttää ei voinut jättää tyhjäksi, testaustulokset olivat identtisiä odotettujen tulosten kanssa.

Miina Ponkala

Mq-valvonnan hallintasovellus, test-case-04:

Purpose: Tiedoston uudelleen nimeäminen tai poistaminen.

Prereq: Käyttäjä on saapunut etusivulle.

Test Data:

Steps:

1. Käyttäjä valitsee navigoinnista joko Confsaver tai Monitoring.
2. Klikkaa delete-kuvaketta.
3. Viestilaatikko kysyy haluatko varmasti poistaa?
4. Jos valitaan OK, tiedosto poistuu, jos Peruuta, palataan takaisin.
5. Toistetaan kohdat 1 ja 2.
6. Klikkaa rename-kuvaketta.
7. Käyttäjä kirjoittaa haluamansa nimen.

Expected results Käyttäjä voi halutessaan poistaa ja/tai nimetä tiedoston uudelleen.

Test results
(date/result/tester) 8.12.2011

Tiedoston poisto ja uudelleen nimeäminen onnistui otetusti.

Miina Ponkala

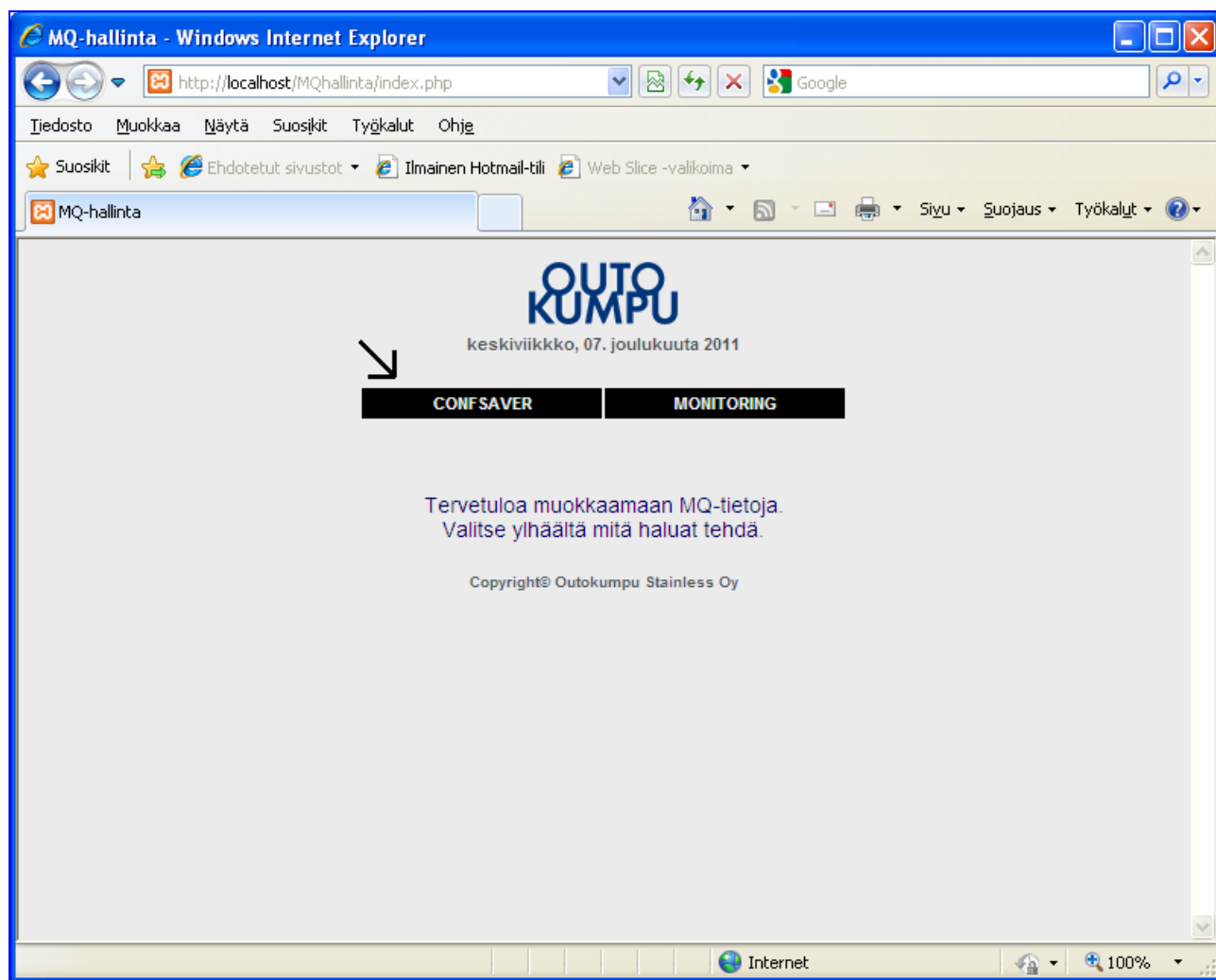
MQ-VALVONNAN HALLINTASOVELLUS

Käyttöohje

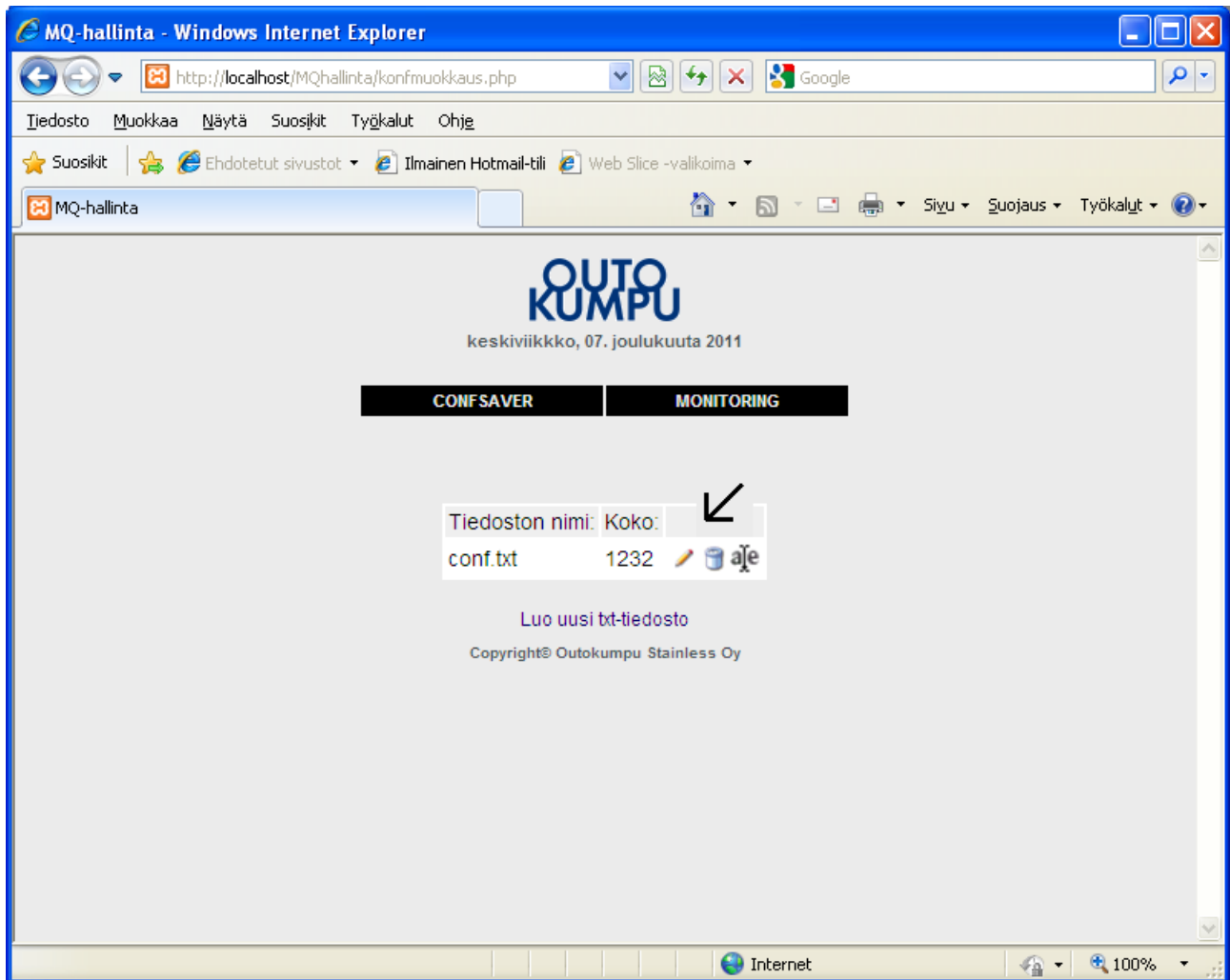
SISÄLTÖ

1	CONFSAVER	Virhe. Kirjanmerkkiä ei ole määritetty.
2	MONITORING	11
3	TIEDOSTOJEN NIMEÄMISKÄYTÄNNÖT	12
4	TIEDOSTOJEN LUONTI	16
5	TIEDOSTOJEN POISTO	16
6.	YLLÄPITO	16
6.1.	Mq-hallinta.....	16
6.2	Inc.....	17
6.3	Inifiles	17
6.4	Confsaver	17
6.5	Js.....	18
6.6	Styles	18

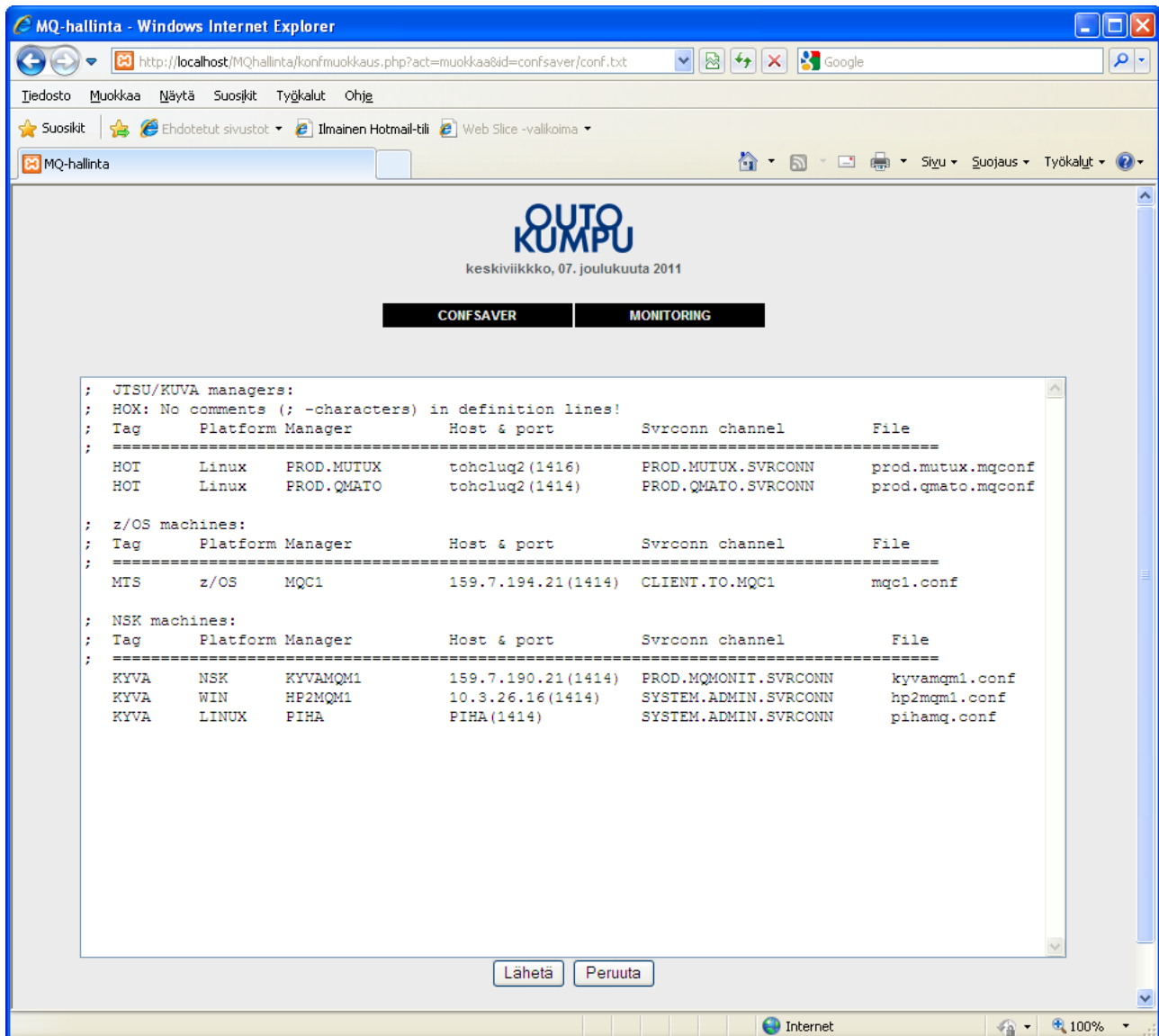
1. CONFSAVER



Valitse etusivulta Confsaver-välilehti. Välilehdelle listautuu kaikki confsaver-hakemistossa sijaitsevat txt-päätteiset tiedostot.



Klikkaa valikosta kynä-ikonia, joka tarkoittaa muokkausta. Tämän jälkeen tullaan varsinaiseen muokkausnäkymään.



Tässä näkymässä voit muokata tietoja vapaasti. Ole huolellinen ja tee muutoksia vain katkoviivojen alapuolella oleviin tietoihin. Lähetä-painikkeesta tiedot tallentuvat palvelimelle. Peruuta-painikkeesta pääsee takaisin alkuun.

















2. MONITORING

Initiedostojen muokkaukseen pääsee samalla tavalla kuin edellä, valitaan vain MONITORING-välilehti alussa. Tiedostot listautuvat kansioittain. Muokkausnäkymään pääsee klikkaamalla kynän kuvaa.

OUTOKUMPU
keskiviikko, 07. joulukuuta 2011

CONFSAVER**MONITORING**

HOT | KYVA | TRC | MTS | PIHA
HOT

Tiedoston nimi:	Koko:	
asd.ini	372	  aļe
dev.mato.ini	353	  aļe
dev.miina.ini	315	  aļe
dev.minttu.ini	345	  aļe
dev.mutux.ini	514	  aļe
dev.qmato.ini	601	  aļe
prod.qpato.ini	371	  aļe
prod.toimi.ini	391	  aļe

Luo uusi ini-tiedosto

Copyright© Outokumpu Stainless Oy

The screenshot displays the Outokumpu Monitoring web interface. At the top, the Outokumpu logo is centered, with the text "keskiviikko, 07. joulukuuta 2011" below it. A navigation bar contains two tabs: "CONFSaver" and "MONITORING". Below this, a row of links includes "HOT", "KYVA", "TRC", "MTS", and "PIHA". The main content area is titled "Connection view" and contains several input fields and a list:

- Queue manager:** A text box containing "KYVAMQM1" with a help icon.
- Platform:** A text box containing "NSK" with a help icon.
- Host:** A text box containing "159.7.190.21(1414)" with a help icon. A yellow tooltip box with the text "Please enter a host or IP." is positioned to the right of this field.
- Server connection:** A text box containing "PROD.MQMONIT.SVRCONN" with a help icon.
- Queues:** A section with a checkbox labeled "Default queue" and a list box containing the following items:
 - PROD.XMIT.QMATO
 - PROD.XMIT.KUTI
 - FROM.AKK
 - FROM.ALP

At the bottom of the form are three buttons: "Send", "Reset", and "Cancel". The footer of the page reads "Copyright© Outokumpu Stainless Oy".

Muokkausnäkylässä on lomake, jossa näkyvät automaattisesti valitun tiedoston connection view – kentät. Hiiren ollessa kysymysmerkin päällä, näkyy ohje kentän täytöstä. Jos Default queue on valittuna, aukeaa alle lomake, johon täytetään jonon oletustiedot.

Platform	<input type="text" value="NSK"/>	?
Host	<input type="text" value="159.7.190.21(1414)"/>	?
Server connection	<input type="text" value="PROD.MQMONIT.SVRCONN"/>	?
Queues		
<input checked="" type="checkbox"/> Default queue		
FROM.MTS	↑	
FROM.MTS	☰	
FROM.MTS1		
FROM.MTS2	↓	
Queue options for default queue		
Alarm limit	<input type="text"/>	?
Email list	<div><div></div></div>	?
Email subject	<input type="text"/>	
Sms list	<input type="text"/>	
Timestamp	<input type="text"/>	?
<div><div>Send</div><div>Reset</div><div>Cancel</div></div>		
Copyright© Outokumpu Stainless Oy		

Kaikki kentät ovat pakollisia. Alarm Täytä kaikki kentät huolellisesti.limit ja Timestamp – kenttien tulee sisältää numeerista tietoa.

3. TIEDOSTOJEN NIMEÄMISKÄYTÄNTÖ

Initiedostojen päähakemisto on Inifiles ja konfiguraatitiedostojen Confsaver. Inifileskansiossa on viisi alihakemistoa, HOT, KYVA, MTS, PIHA ja TRC. Muokattavat tiedostot sijaitsevat osastokohtaisesti näissä alihakemistoissa. Confsaver-hakemistossa ei ole alikansioita, vaan tiedostot sijaitsevat suoraan sen juuressa.

Kun luodaan uusi tiedosto tai nimetään vanha tiedosto uudelleen, on nimen alkuun laitettava kansio ja päätteeksi.ini tai txt riippuen siitä kumpaan kategoriaan tiedosto luodaan.

Esim. uusi ini-tiedosto HOT kansioon on muotoa HOT/prod.qmato.ini.

4. TIEDOSTOJEN LUONTI

4.1 Monitoring

OUTOKUMPU

















keskiviikko, 07. joulukuuta 2011


CONFSAVER

MONITORING

HOT | KYVA | TRC | MTS | PIHA

HOT

Tiedoston nimi:	Koko:	
asd.ini	372	  aje
dev.mato.ini	353	  aje
dev.miina.ini	315	  aje
dev.minttu.ini	345	  aje
dev.mutux.ini	514	  aje
dev.qmato.ini	601	  aje
prod.qpato.ini	371	  aje
prod.toimi.ini	391	  aje

 [Luo uusi ini-tiedosto](#)

Copyright© Outokumpu Stainless Oy

Initiedostojen listauksen alapuolella näkyy linkki, jota klikkaamalla pääsee luomaan uuden tiedoston.

HOT | KYVA | TRC | MTS | PIHA

Tiedoston nimi:
E.g. HOT/prod.qmato.ini

New form

Queue manager

Platform

Host

Server connection

Queues

Alarm limit

Email list

Email subject

Sms list

Timestamp

Lähetä

Peruuta

Uuden tiedoston luomisessa täytetään huolellisesti kaikki kentät. Hälytystiedot tallentuvat automaattisesti oletusjonolle (Default queue). Save-painike tallentaa tiedoston nimessä annettuun kansioon.

4.2. Confsaver

Konfiguraatiopuolella toimitaan muuten samoin, mutta luontinäkö on erilainen.

OUTO KUMPU
keskiviikko, 07. joulukuuta 2011

CONFSAVER **MONITORING**

Tiedoston nimi: Muista tiedoston pääte! (.txt)

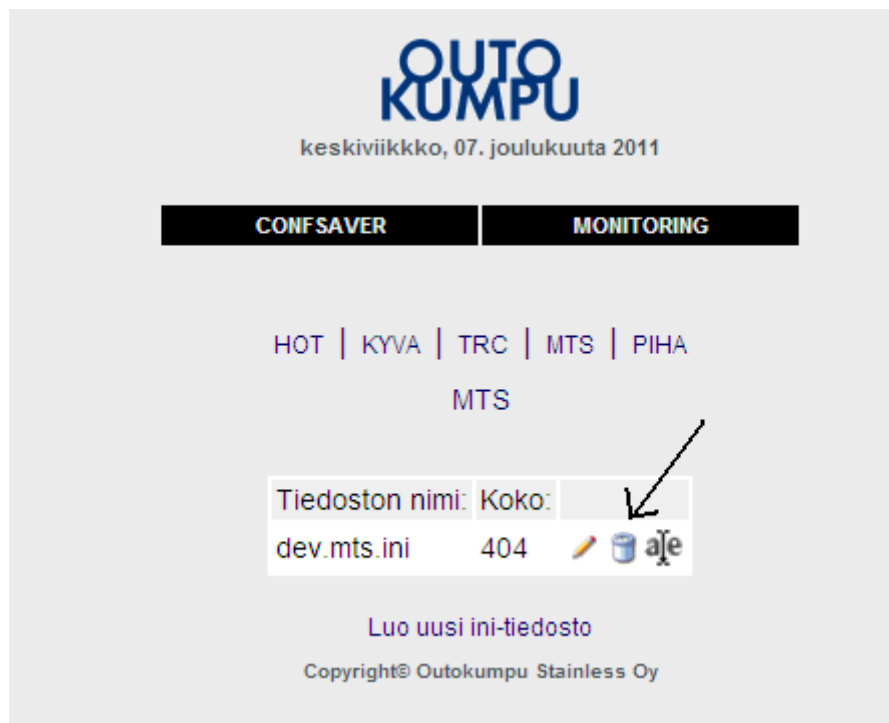
Teksti:

```
; JTSU/KUVA managers:  
; HOX: No comments (; -characters) in definition lines!  
; Tag      Platform Manager      Host & port      Svrconn channel      File  
; =====  
  
; z/OS machines:  
; Tag      Platform Manager      Host & port      Svrconn channel      File  
; =====  
  
; NSK machines:  
; Tag      Platform Manager      Host & port      Svrconn channel      File  
; =====
```

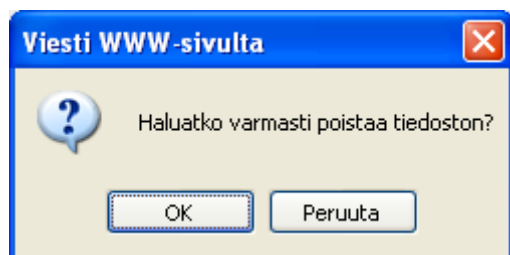
Uusi conf-tiedosto luodaan kuten muokkausnäkössä, kokonaisuudessaan kuten sen halutaan näkyvän itse tiedostossa. Halutut tiedot lisätään katkoviivojen alapuolelle.

5. TIEDOSTON POISTO

Käyttäjän valittua MONITORING tai CONFSAVER välilehden, tiedostot listautuvat kuten kuvassa alla. Jokaisen tiedoston nimen perässä on roskakorin kuva, jota klikkaamalla tiedosto poistetaan.



Ennen poistamista varmistetaan, halutaanko valittu tiedosto varmasti poistaa.



Jos valitaan OK, tiedosto poistuu. Peruuta-painike palauttaa käyttäjän alkuun.

6. YLLÄPITO

6.1. MQhallinta

MQhallinta on sovelluksen päähakemisto, jonka juuressa ovat tiedostot index.php, muokkaus.php ja konfmuokkaus.php. Hakemisto sisältää myös kuusi alihakemistoa, joiden sisältö mainitaan alempana.

Index.php on sovelluksen etusivu, joka sisältää ylä- ja alaindeksin sekä valinnaista tekstiä.

Muokkaus.php on Monitoring-välilehden pääsivu, jossa kutsutaan kaikkia tarvittavia funktioita.

Konfmuokkaus.php sisältää Confsaver-välilehden vastaavat tiedot.

6.2. Inc

Inc-kansio on MQhallinnan alihakemisto, joka sisältää muokkaussivuille linkitettävät tiedostot. Funktiot.php ja Confunktiot.php sisältävät nimensä mukaisesti kaikki sovelluksessa käytettävät funktio, kuten listaus, muokkaus ja uuden tiedoston luonti.

Yla.php on sivuston yläindeksi, joka sisältää päivämääräasetukset, navigaatiomodulin ja tyyli-tiedoston sekä js-tiedostojen linkityksen. Ala.php on alaindeksi joka sisältää tekijänoikeustiedot. Ylä- ja alaindeksit näkyvät kaikilla sivuilla.

6.3. Inifiles

Kaikki ini-päätteiset tiedostot sijaitsevat inifileshakemiston alihakemistoissa, joita ovat TRC, PIHA, KYVA, MTS ja HOT. Tiedostot on jaoteltu osastokohtaisesti.

6.4 Confsaver

Hakemisto sisältää Confsaver -näkylässä muokattavan conf.txt tiedoston.

6.5. Js

Js -hakemisto sisältää sovelluksessa käytettävät JavaScript –koodit. addInput.js sisältää lomakkeen kenttien tarkistuskoodit ja control.js sisältää koodipätkän, jolla piilotettu Options-lomake tuodaan esiin.

6.6 Styles

Styles hakemisto sisältää kaksi tiedostoa. Style.css sisältää kaikki sovelluksessa käytettävät tyylimäärittelyt. Csshover on HTC-tiedosto, joka sisältää tyylimäärittelyjä navigointipalkin mahdollista laajentamista varten.